Journal of Electronics and Information Science (2023)
Clausius Scientific Press, Canada

DOI: 10.23977/jeis.2023.080610
ISSN 2371-9524 Vol. 8 Num. 6

An Efficient Lidar-Based Algorithm for Autonomous Vehicle's Visual Detection

Caixia Zhao[1,a,*], Yu Zhang[2,b]

[1]School of Air Transportation and Engineering, Nanhang Jincheng College, Nanjing, China
[2]School of Electronic Science and Engineering, Nanjing University, Nanjing, China
[a]zhaocx_atm@nuaa.edu.cn, [b]yzhang@topxgun.com
[*]Corresponding author

Keywords: Autonomous driving, deep learning, vehicle detection, lidar point cloud

Abstract: The real-time and accurate three-dimensional object detection is one of the core tasks in the perception of autonomous driving environments. In recent years, the development of deep learning technology and lidar technology has led to significant advancements in the application of three-dimensional object detection algorithms in large-scale general scenarios. However, existing lidar-based three-dimensional object detection algorithms still face challenges in complex traffic scenarios, and the difficulty lies in balancing the accuracy and inference speed of the algorithms. In this regard, the voxel-based single-stage three-dimensional object detection algorithm SECOND is used as the baseline algorithm and an efficient single-stage vehicle detection algorithm framework tailored for complex autonomous driving scenarios is proposed. Firstly, a residual structure is introduced and the feature channel number is reconstructed in the three-dimensional feature extraction backbone, which effectively reduce the loss of spatial geometric features in the point cloud during the feature extraction process and make the model training more stable. Secondly, the multi-scale feature fusion technology and a spatial feature attention mechanism are introduced and a more efficient two-dimensional feature fusion backbone is designed, which facilitates the learning of the model for vehicle size and orientation. The proposed algorithm is trained and validated on the open-source dataset ONCE. Compared to the baseline algorithm, the average detection accuracy for vehicles is improved by 5.64%, while maintaining an inference speed of 20 frames per second (FPS). This significantly enhances the algorithm's perception performance for vehicles in complex traffic scenarios.

1. Introduction

The perception algorithms provide necessary environmental information for downstream tasks such as decision-making and control in autonomous driving vehicles. The purpose of three-dimensional object detection is to obtain information about the position, size, orientation, and other attributes of objects in the three-dimensional world. It is one of the core technologies in the environmental perception of autonomous driving vehicles and is crucial for enhancing driving safety [1]. In recent years, with the widespread application and continuous cost reduction of lidar in

the fields of autonomous driving and robotics, lidar-based three-dimensional object detection algorithms have been widely researched in both industry and academia. Compared to sensors such as in-vehicle cameras and millimeter-wave radar, lidar can directly acquire depth information and is not affected by lighting conditions, providing the highest accuracy and robustness [2]. Therefore, lidar-based vehicle detection algorithms have great research significance.

The point cloud detection algorithms based on deep learning can be broadly categorized into three main technical approaches: methods based on raw points, methods based on voxels, and hybrid methods combining raw points and voxels. Methods based on raw points involve direct use of neural networks for feature extraction from the raw point cloud. Specifically, techniques such as multi-layer perceptrons and max-pooling layers are employed for point cloud feature extraction. Pioneering works such as PointNet [3] and PointNet++ [4] use neural networks to extract features from the raw point cloud, effectively capturing global and local spatial geometric features in point clouds, laying the foundation for subsequent raw point-based point cloud detection algorithms. Directly using neural networks for feature extraction from raw point clouds significantly limits the algorithm's inference speed, making it difficult to meet the real-time requirements of autonomous driving systems. For this, voxel-based methods have emerged. The core idea is to first partition irregular point cloud data in the perception scene into regular grids and then use two-dimensional or three-dimensional convolutions for feature extraction. VoxelNet [5] is the first work to propose partitioning the raw point cloud into a three-dimensional voxel grid and using three-dimensional convolutions for feature extraction. Its designed voxel feature encoding layer has been widely adopted in subsequent works. Considering that applying general three-dimensional convolutions directly to voxelized point clouds introduces a considerable computational burden, Yan Y et al. introduced three-dimensional sparse convolutions [6] and submanifold convolutions [7] to construct a three-dimensional sparse feature extraction backbone for extracting three-dimensional features. They proposed an efficient real-time point cloud detection algorithm, SECOND [8], tailored to autonomous driving scenarios, becoming a classic paradigm for subsequent voxel-based point cloud detection algorithms. Considering that methods based on raw points effectively preserve fine geometric features in point clouds, and voxel-based methods with voxel encoding are computationally friendly, hybrid methods combining raw points and voxels adopt a mixed architecture. PV-RCNN [9] is the most classic algorithm in this architecture. It first utilizes the SECOND algorithm as a one-stage method to obtain rough target candidate boxes. Then, using the farthest point sampling algorithm, it selects several key points from the raw point cloud. Subsequently, it aggregates intermediate layer features from three-dimensional feature extraction backbones and pseudo-bird's-eye-view features to key points. Finally, it refines the target candidate boxes obtained in the one-stage method using key point features, resulting in more accurate target bounding boxes.

Based on the presence or absence of a two-stage refinement module, current point cloud detection algorithms can be classified into single-stage detection algorithms and two-stage detection algorithms. Single-stage algorithms often have higher inference speeds and relatively lower detection accuracy compared to two-stage algorithms, examples include SECOND, PointPillars [10], etc. In contrast, two-stage detection algorithms, due to the use of a two-stage refinement module, generally exhibit higher detection accuracy and reduced inference speed. Examples of two-stage algorithms include PV-RCNN, Voxel RCNN [11], CT3D [12], etc. Considering the real-time requirements of autonomous driving, SECOND is chosen as the baseline algorithm in this work. The car is one of the most prevalent participants in traffic scenarios, particularly in driving scenarios such as highways, which directly influences the decision-making and planning of autonomous vehicles. While the existing SECOND detection algorithm achieves good detection results in simpler driving scenarios, it still faces challenges in detecting vehicles in complex traffic

scenarios. To address this, the present study optimizes and improves the SECOND algorithm, resulting in a vehicle detection algorithm that is better suited for complex traffic scenarios.

Due to the inherent properties of lidar, the point cloud scanned on the surface of a vehicle is highly incomplete and discontinuous. Moreover, this characteristic worsens with an increase in detection distance. This necessitates that detection algorithms maximize the utilization of target point cloud features during the feature extraction process. The existing SECOND algorithm has a limitation in the three-dimensional feature extraction backbone, as it does not sufficiently extract point cloud features, leading to a significant loss of fine geometric features. This limitation is a crucial bottleneck restricting the improvement of vehicle detection performance. To address this, the paper introduces a residual structure, which effectively reduces feature loss during the feature extraction process in the three-dimensional feature extraction backbone. The introduction of the residual structure also facilitates easier convergence of the algorithm. Additionally, the paper reconstructs the number of feature channels in the three-dimensional feature extraction backbone to make it more suitable for complex traffic scenarios. The role of the two-dimensional feature fusion backbone is to further integrate features from the three-dimensional feature extraction backbone to obtain the final features used for target detection. To ensure that the two-dimensional feature fusion backbone obtains more features favorable for vehicle localization, the paper introduces multiscale feature fusion technology and a spatial feature attention mechanism, designing a more effective two-dimensional feature fusion backbone. To validate the effectiveness of the proposed algorithm, experiments and validation are conducted on the ONCE dataset [13] collected in China.

2. Vehicle Detection Method Design

2.1. Framework Overview

The SECOND method is a single-stage anchor-based point cloud detection algorithm. In this paper, it serves as the baseline algorithm, and optimizations are conducted around its network structure. The proposed network architecture includes point cloud voxel encoding, three-dimensional feature extraction backbone, two-dimensional feature extraction backbone, and the detection head. The architecture of the proposed algorithm is illustrated in Figure 1.

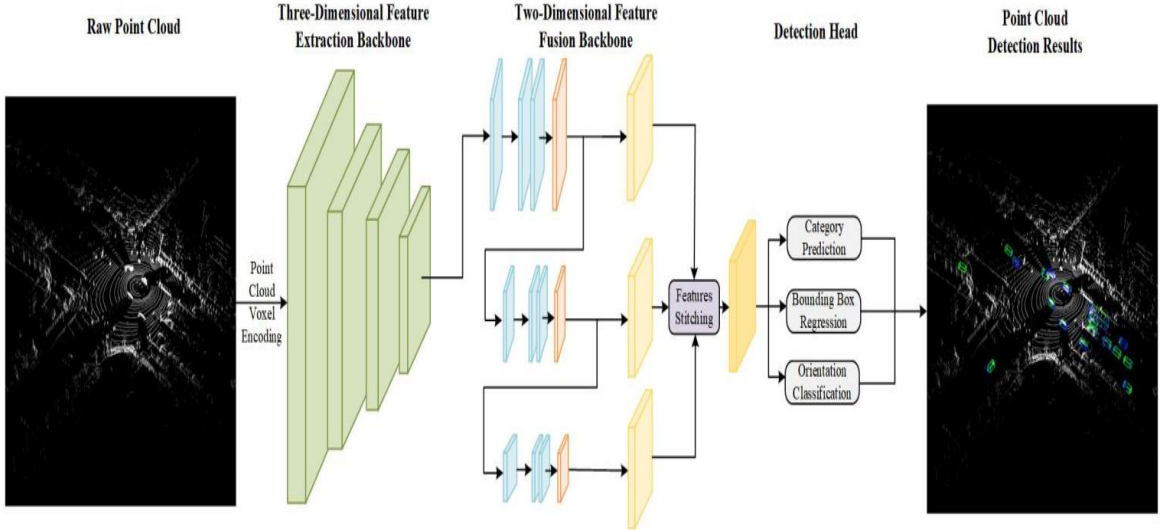

Figure 1: Algorithmic network architecture.

2.2. Voxel Feature Encoding

Each point in the input point cloud is represented by three-dimensional coordinates (x, y, z) and reflectance intensity. Based on a predetermined voxel size, the input point cloud is segmented into voxels along the X, Y, and Z axes. Subsequently, each point in the point cloud is assigned to the corresponding voxel based on its coordinate position. The next step involves calculating the average feature value for all points within the same voxel, which serves as the voxel-level feature.

2.3. Three-Dimensional Feature Extraction Backbone

The role of the three-dimensional feature extraction backbone is to further extract features from the output of the voxel feature encoding module. The feature extraction capability of this module significantly impacts the subsequent effectiveness of object detection. To achieve vehicle detection in complex traffic scenarios, the three-dimensional feature extraction backbone needs to capture more discriminative geometric features from the point cloud. Considering the severe feature loss issue in the original three-dimensional feature extraction backbone of SECOND and the difficulty of its extracted point cloud features in meeting the requirements of vehicle detection in complex scenes, this paper introduces a more efficient three-dimensional feature extraction backbone by incorporating a residual structure [14] and reconstructing the feature channel number, as illustrated in Figure 2.

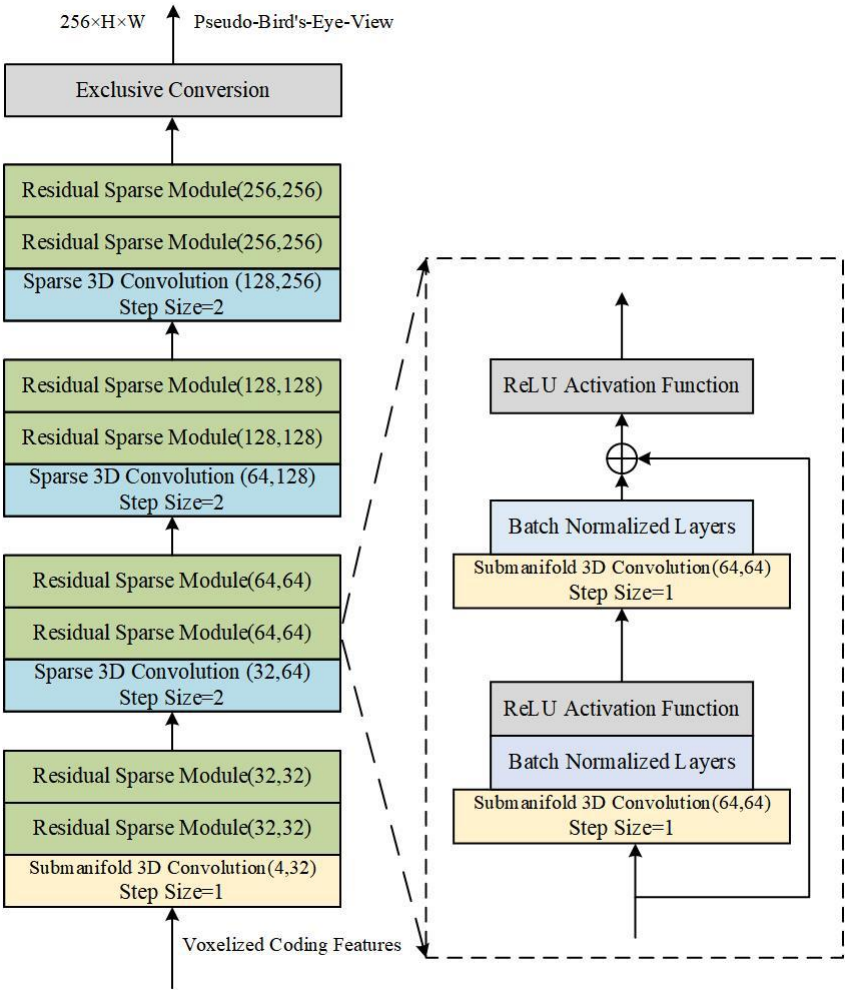

Figure 2: Three-Dimensional Feature Extraction Backbone.

Similar to SECOND, the designed three-dimensional feature extraction backbone in this paper consists of four stages, performing 1×, 2×, 4×, and 8× downsampling, with feature channel numbers of 32, 64, 128, and 256 at each stage, ultimately outputting two-dimensional pseudo-bird's-eye-view features. The specific structure of the residual sparse module is shown on the right side of Figure 2. Compared to the original three-dimensional feature extraction backbone, the constructed three-dimensional feature extraction backbone in this paper can extract finer geometric features from the point cloud, significantly reducing feature loss, and effectively capturing more discriminative point cloud features for complex traffic scenarios.

2.4. Two-Dimensional Feature Fusion Backbone

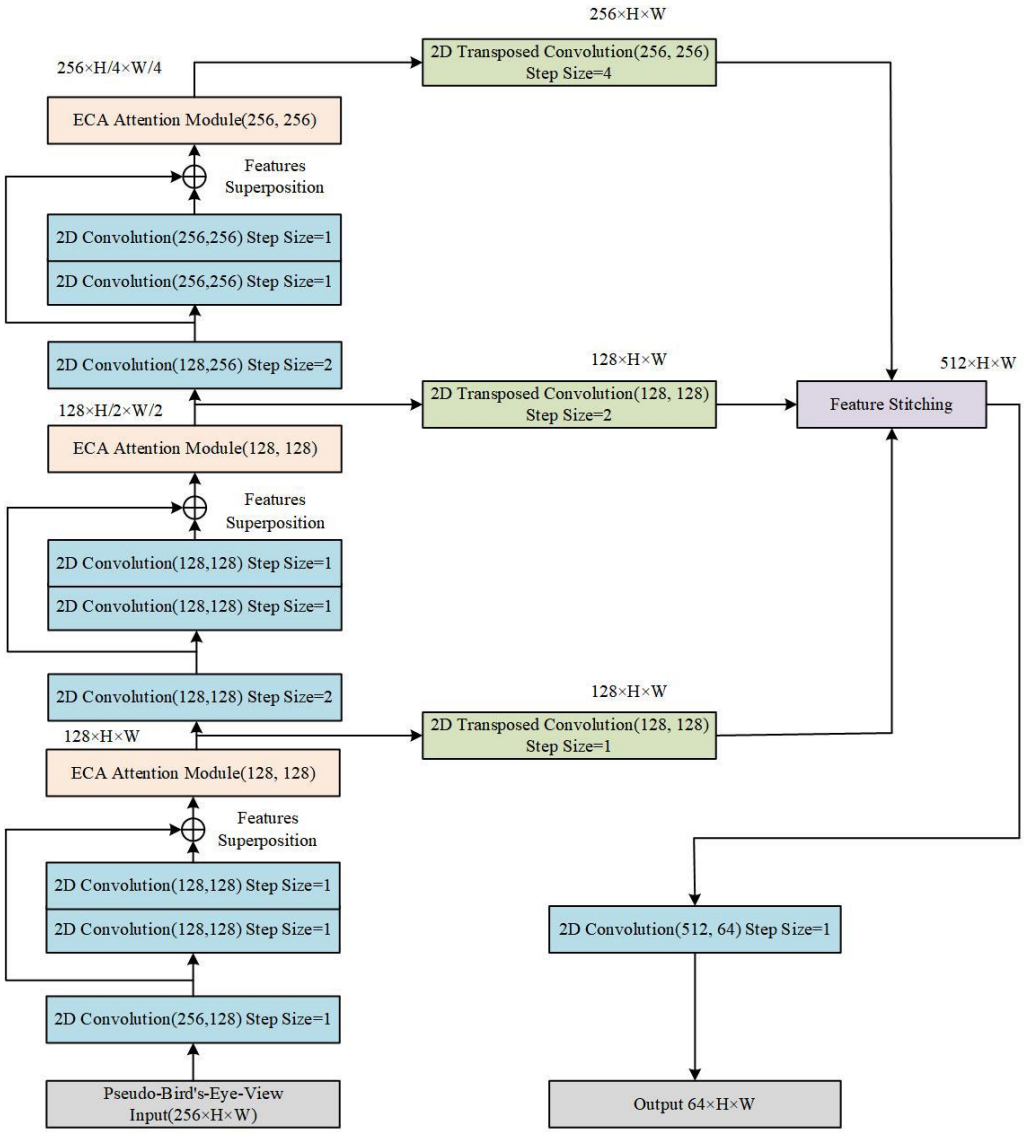

Figure 3: Two-Dimensional Feature Fusion Backbone.

The two-dimensional feature fusion backbone takes the pseudo-bird's-eye-view features output by the three-dimensional feature extraction backbone as input and further integrates these features to obtain the final features used for vehicle detection. Among various traffic participants, vehicles have larger dimensions compared to pedestrians and cyclists. Therefore, the point cloud scanned on the surface of a vehicle contains the highest number of points. However, due to the limitations of

lidar's angular resolution, the number of laser beams, and occurrences of occlusion, the scanned point cloud on the vehicle surface is often incomplete and discontinuous. This characteristic has historically resulted in suboptimal performance of vehicle detection algorithms in scenarios where point cloud data is severely missing, leading to inaccurate predictions of vehicle size and orientation angles.

In complex traffic scenarios, occlusion situations are frequent, imposing higher demands on the two-dimensional feature fusion backbone. To achieve accurate predictions of vehicle size and orientation, the features output by the two-dimensional feature fusion backbone need to include more target boundary features. To address this, the paper introduces a more efficient two-dimensional feature fusion backbone, as shown in Figure 3. To fuse more comprehensive point cloud features, the proposed two-dimensional feature fusion backbone adopts a three-tiered design, fully integrating the features extracted at these three levels. Additionally, to direct the network's attention to more vehicle boundary features, the paper introduces the ECA attention mechanism [15], which effectively enhances the algorithm's accuracy in locating vehicles. Compared to the original two-dimensional feature fusion backbone, the designed two-dimensional feature fusion backbone in this paper can obtain more discriminative point cloud features.

2.5. Detection Head

The role of the detection head is to perform the final detection of vehicles, including category prediction and bounding box parameter regression. Consistent with SECOND, this paper employs an anchor-based detection head. The detection head comprises three sub-heads: category prediction, bounding box regression, and orientation classification. The orientation classification sub-head is introduced to further improve the algorithm's accuracy in predicting the orientation angles of vehicles. Based on the information predicted by these three sub-heads, the actual three-dimensional bounding box of the vehicle can be decoded.

2.6. Loss Functions

In order to facilitate rapid convergence of the network, this paper employs a series of loss functions, primarily including classification loss L_{cls}, bounding box regression loss L_{reg}, and orientation classification loss L_{dir}. The classification loss uses the Focal loss function, the bounding box regression loss uses the Smooth L1 loss function, and the orientation classification loss uses the Cross-Entropy loss function. The final total loss L_{total} is defined as:

$$L_{total} = \lambda_{cls} L_{cls} + \lambda_{reg} L_{reg} + \lambda_{dir} L_{dir}$$

Where λ_{cls}, λ_{reg}, λ_{dir}, represent the weights of classification loss, bounding box regression loss, and orientation classification loss.

3. Experimental verification

3.1. Introduction to Public Dataset

The ONCE dataset was collected by Huawei in China and encompasses various weather conditions (clear, cloudy, rainy, etc.), different time periods (morning, noon, afternoon, night), and diverse road conditions (downhill, suburban, highway, tunnel, bridge, etc.). The dataset is captured using seven cameras and a 40-line lidar. Within the ONCE dataset, 5,000, 3,000, and 8,000 frames of point cloud data are separately annotated for training, validation, and testing purposes. The

labeled target categories include cars, trucks, buses, pedestrians, and cyclists. In this paper, cars, trucks, and buses are collectively categorized as vehicles.

3.2. Implementation Details

In this work, an end-to-end approach is employed to train the vehicle detection model. We utilize two Nvidia RTX 3090 graphics cards to train our network for 80 epochs, with a batch size set to 16. Additionally, we use the AdamW optimizer [16] and adopt a one-cycle learning rate optimization strategy, where the maximum learning rate is set to 1e-3, weight decay is 0.01, and momentum ranges from 0.85 to 0.95. Furthermore, in this study, the point cloud detection range is set to [75.2m, 75.2m] along both the X and Y axes, and [-5m, 3m] along the Z-axis. The voxel size is (0.1m, 0.1m, 0.2m).

Table 1: The results of the improved algorithm are compared with those of other algorithms.

Method	Source	Sensor	Single Stage/ Two-Stage	Car(AP)
PointPillars	CVPR2019	Lidar	Single Stage	68.57
PV-RCNN	CVPR2020	Lidar	Two-Stage	77.77
VoxelRCNN	AAAI2021	Lidar	Two-Stage	77.48
Part-A2-Anchor	TPAMI2020	Lidar	Two-Stage	76.91
CT3D	ICCV2021	Lidar	Two-Stage	78.66
CenterPoints	CVPR2021	Lidar	Single Stage	66.79
SECOND	Sensors2018	Lidar	Single Stage	71.19
The proposed algorithm	-	Lidar	Single Stage	76.83

Table 2: The results of the improved algorithm are compared with those of the benchmark algorithm.

Method	Car			
	0-30m	30-50m	50-inf	AP
SECOND	84.04	63.02	47.25	71.19
The Improved Algorithm	86.62	70.25	56.4	76.83
Performance Improvement	+2.58	+7.23	+9.15	+5.64

Table 3: Ablation experiments on the ONCE validation set.

Method	Car				FPS
	0-30m	30-50m	50-inf	AP	
SECOND	84.04	63.02	47.25	71.19	25
SECOND+Residual 3D Trunk	85.96	69.95	55.08	75.93	22
SECOND+Residual 3D Trunk+Multi-scale Spatial Feature Fusion Module	86.62	70.25	56.4	76.83	20

3.3. Experimental Results

In this section, we compare the experimental results of the proposed model on the open-source ONCE dataset with baseline algorithms and some advanced 3D object detection algorithms. Table 1 presents a comparison of the experimental results before and after the proposed algorithm, revealing a noticeable improvement in the average precision (AP) of vehicle detection by 5.64% compared to

the baseline algorithm. At distance scales of 0-30m, 30-50m, and 50-inf, the proposed algorithm demonstrates improvements of 2.58%, 7.23%, and 9.15%, respectively. This indicates a significant enhancement in the detection accuracy of mid-to-long-range targets.

Additionally, we compare the proposed vehicle detection algorithm with some mainstream advanced algorithms, as shown in Table 2. From Table 2, it can be observed that the average detection accuracy of the proposed vehicle detection algorithm is 8.26%, 10.04%, and 5.64% higher than popular single-stage detection algorithms such as PointPillars, CenterPoints [17], and SECOND. Moreover, the proposed algorithm substantially narrows the gap with two-stage detection algorithms, being only 0.94%, 0.65%, 0.08%, and 1.83% lower than PV-RCNN, Voxel RCNN, Part-A2-Anchor [18], and CT3D, respectively.

For a more intuitive representation of the advantages of the proposed vehicle detection algorithm over the baseline, Figure 4 provides a visual comparison of vehicle detection results. From top to bottom, the images depict the ground truth, baseline algorithm detection results, and improved algorithm detection results, respectively. Red circles indicate false positives, orange circles represent false negatives. It is evident that the improved algorithm exhibits significantly fewer false positives and false negatives compared to the baseline algorithm, further highlighting the effectiveness of the proposed algorithm.

3.4. Ablation Experiments

To verify the effectiveness of each component of the proposed method, we conducted ablation experiments testing each improvement on the ONCE validation set. Table 3 presents the experimental results. From Table 3, it is evident that the adoption of the residual 3D backbone results in a 4.74% improvement in the average detection accuracy compared to the baseline algorithm. At distance scales of 0-30m, 30-50m, and 50-inf, the detection accuracy increases by 1.92%, 6.93%, and 7.83%, respectively. This indicates that the improved residual 3D backbone effectively promotes the network's learning of fine-grained point cloud geometric features, significantly reducing feature loss during the extraction process.

Moreover, upon incorporating the designed multi-scale spatial feature fusion module on top of the residual 3D backbone, the algorithm's average detection accuracy for vehicle detection improves by 0.9% compared to the baseline algorithm. At distance scales of 0-30m, 30-50m, and 50-inf, the detection accuracy increases by 0.66%, 0.3%, and 1.32%, respectively. This suggests that the designed 2D feature fusion backbone effectively learns boundary point cloud features of vehicles, enhancing the precision of vehicle localization.

Additionally, from Table 3, it is observed that the improved vehicle detection results in a reduction in the inference speed from the original 25 FPS to 20 FPS. This slight decrease of 5 FPS still satisfies the real-time requirements of autonomous vehicles.

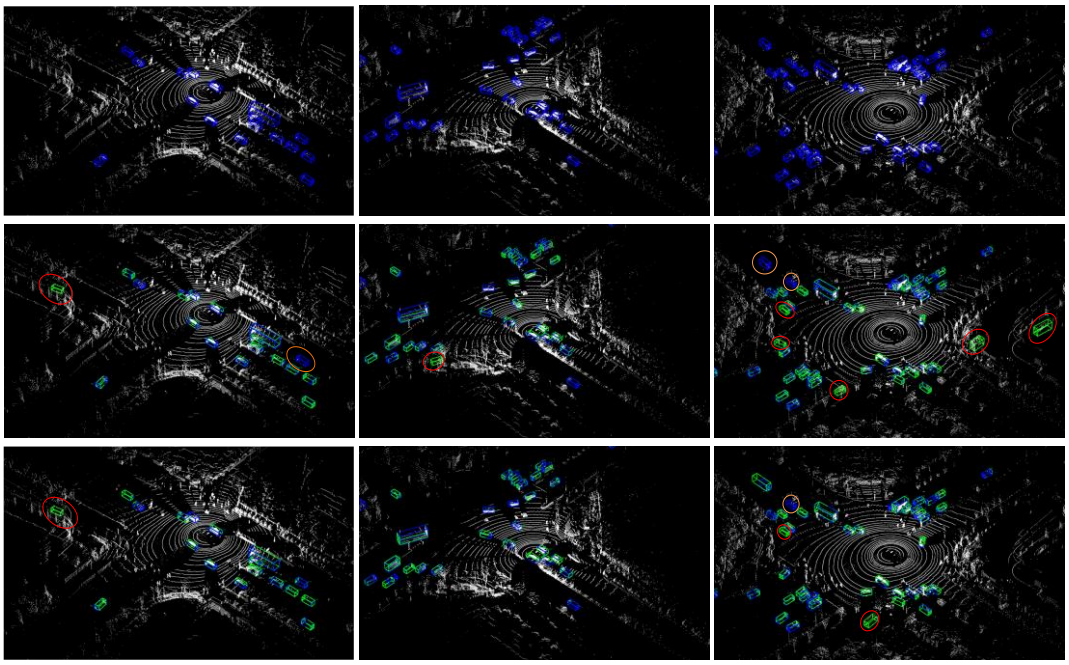

Figure 4: ONCE verify a visual comparison plot on a set.

4. Conclusions

This paper proposes a novel single-stage anchor-based vehicle detection algorithm tailored for complex traffic scenarios, using the SECOND algorithm as a baseline. By introducing a residual structure and reconstructing feature channel numbers, a more effective three-dimensional feature extraction backbone is constructed. This approach successfully mitigates feature loss during extraction, preserving intricate spatial geometric features of point clouds. Additionally, through the application of multi-scale feature fusion techniques and spatial feature attention mechanisms, a more efficient two-dimensional feature fusion backbone is designed, further enhancing the algorithm's precision in vehicle localization. Experimental validation on the ONCE open-source dataset demonstrates that, compared to the baseline algorithm, the improved algorithm achieves a 5.64% increase in the average detection accuracy for vehicles while maintaining an algorithmic inference speed of 20 FPS.

References

[1] Jing Qin, Weibin Wang, Qijie Zou, et al. Reviewof 3D Target Detection Methods Based on LiDAR Point Cloud [J]. Computer Science, 2023, 50(S1):259-265.

[2] Zhenqi Wei. Research on Lidar-based 3D Object Detection Algorithm for Intelligent Vehicles [D].Jilin University, 2023. DOI:10.27162/d.cnki.gjlin.2023.001093.

[3] Qi C R, Su H, Mo K, et al. Pointnet: Deep learning on point sets for 3d classification and segmentation[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 652-660.

[4] Qi C R, Yi L, Su H, et al. Pointnet++: Deep hierarchical feature learning on point sets in a metric space [J]. Advances in neural information processing systems, 2017, 30.

[5] Zhou Y, Tuzel O. Voxelnet: End-to-end learning for point cloud based 3d object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 4490-4499.

[6] B. Graham, "Sparse 3D convolutional neural networks," 2015, arXiv:1505.02890.

[7] B. Graham and L. van der Maaten, "Submanifold sparse convolutional networks," 2017, arXiv:1706.01307.

[8] Yan Y, Mao Y, Li B. Second: Sparsely embedded convolutional detection [J]. Sensors, 2018, 18(10): 3337.

[9] Shi S, Guo C, Jiang L, et al. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10529-10538.

[10] Lang A H, Vora S, Caesar H, et al. Pointpillars: Fast encoders for object detection from point clouds[C]// Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 12697-12705.
[11] Deng J, Shi S, Li P, et al. Voxel r-cnn: Towards high performance voxel-based 3d object detection[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2021, 35(2): 1201-1209.
[12] Sheng H, Cai S, Liu Y, et al. Improving 3d object detection with channel-wise transformer[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 2743-2752.
[13] Mao J, Niu M, Jiang C, et al. One million scenes for autonomous driving: Once dataset[J]. arXiv preprint arXiv: 2106.11037, 2021.
[14] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
[15] Wang Q, Wu B, Zhu P, et al. ECA-Net: Efficient channel attention for deep convolutional neural networks[C]// Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 11534-11542.
[16] Loshchilov I, Hutter F. Decoupled weight decay regularization [J]. arXiv preprint arXiv:1711.05101, 2017.
[17] Yin T, Zhou X, Krahenbuhl P. Center-based 3d object detection and tracking[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 11784-11793.
[18] Shi S, Wang Z, Wang X, et al. Part-aˆ 2 net: 3d part-aware and aggregation neural network for object detection from point cloud [J]. arXiv preprint arXiv:1907.03670, 2019, 2(3).